REMARKS

Applicant is in receipt of the Office Action mailed April 23, 2004. Claims 1 – 33 were rejected. New claims 34 and 35 have been added. Claims 1 - 35 are currently pending in the application.

Applicant notes that a new oath or declaration is required because a provisional application number listed on the original was incorrect. Applicant is in the process of procuring a new signed declaration, and will provide this declaration as soon as it is available.

The specification was objected to because a provisional application number provided in the priority claim added in the first Preliminary Amendment was incorrect. Applicant has amended the priority claim to use the correct provisional application number.

Claim Rejections – 35 U.S.C. 112

Claim 21 was rejected under 35 U.S.C. 112 because it included the limitation, "The memory medium of claim 10," but claim 10 is a method claim. Applicant has amended claim 21 to correct it so that it instead includes the limitation, "The memory medium of claim 20".

Claim Rejections – 35 U.S.C. 102

Claims 1 – 10 and 20 – 27 were rejected under 35 U.S.C. 102(b) as being anticipated by U.S. Patent No. 5,987,246 to Thomsen et al. (hereinafter referred to as "Thomsen"). Applicant respectfully traverses this rejection.

Thomsen relates generally to a graphical programming system and method which includes three-dimensional nodes that are wired or connected to form a graphical program or block diagram. Applicant submits that there are numerous elements of the present claims that Thomsen does not teach or suggest.

As per claim 1, the claim recites in pertinent part:

> receiving user input specifying configuration information for the node;
>
> programmatically creating and displaying one or more input terminals and one or more output terminals for the node, based on the configuration information;

Regarding the element of "programmatically creating and displaying one or more input terminals and one or more output terminals for the node, based on the configuration information", the Examiner simply states "(inherent)" but does not provide any reasoning as to why he believes this to be inherent in Thomsen's teaching. Applicant submits that this feature is not inherent in Thomsen's teaching, and in fact, Thomsen's system is not capable of programmatically creating and displaying input terminals or output terminals for a node based on configuration information specified by user input.

As noted on p.44 of the present application, an input terminal for a node comprises a connection point where the node can receive data from a data source in the graphical program (e.g., from an output terminal of another node). An output terminal for a node comprises a connection point where the node can provide data to a data target in the graphical program (e.g., to an input terminal of another node). Thomsen contains no teaching or suggestion, either inherent or otherwise, regarding an ability to programmatically create and display input terminals or output terminals for a node based on configuration information for the node specified by user input.

Applicant thus submits that claim 1 is allowable. Inasmuch as claims 20 and 34 recite similar features as those of claim 1, Applicant submits that these claims are also allowable. Since claims 2 – 12 depend on claim 1 and claims 21 – 29 depend on claim 20, Applicant submits that these claims are also allowable for at least this reason. Additionally, Applicant submits that dependent claims 2 – 12 and 21 – 29 recite numerous additional limitations not taught or suggested by Thomsen.

For example, claim 3 recites:

> 3. The method of claim 1,
>
> wherein said receiving user input specifying the configuration information comprises receiving user input specifying:
>
> one or more input terminals from a set of possible input terminals; and
>
> one or more output terminals from a set of possible output terminals;

> wherein said programmatically creating and displaying one or more input terminals and one or more output terminals for the node comprises programmatically creating and displaying the one or more input terminals and the one or more output terminals specified by the user input.

Regarding claim 3, the Office Action states, "Thomsen teaches one or more input terminals and one or more output terminals (Fig. 9 – 10), wherein creating and displaying the one or more input terminals and the one or more output terminals specified by the user input (col. 7, lines 18 – 29)." Applicant respectfully submits that this is not a grammatically understandable sentence and does not clearly explain the Examiner's reasoning for rejecting claim 3.

Applicant also submits that Thomsen does not teach or suggest the combination of elements recited in claim 3. The cited Figures 9 – 10 of Thomsen simply illustrate examples of graphical programs which include three-dimensional nodes. The cited portion of Col. 7, lines 18 – 29 generally relates to a user assembling a graphical program by interconnecting the three-dimensional node icons. There simply is no teaching or suggestion whatsoever in the cited portion regarding receiving user input specifying one or more input terminals from a set of possible input terminals and programmatically creating and displaying the one or more input terminals specified by the user input. Similarly, there is no teaching or suggestion regarding receiving user input specifying one or more output terminals from a set of possible output terminals and programmatically creating and displaying the one or more output terminals specified by the user input. Applicant thus submits that claim 3 is allowable.

Taking claim 9 as another exemplary claim, the claim recites the method of claim 1, further comprising: "programmatically generating graphical source code for the node to implement functionality specified by the configuration information." Applicant first notes that the configuration information recited in claim 9 comprises information specified by user input (since claim 9 depends on claim 1). However, Thomsen does not even teach the concept of user-specified configuration information which specifies functionality for a node. Furthermore, Thomsen certainly does not teach the concept of programmatically generating graphical source code for a node to implement functionality specified by configuration information, where the configuration information is specified by user input. The Office Action relies on Col. 8, lines 50 – 59 of Thomsen to teach claim 9. However, this portion of Thomsen generally relates to the system providing an indication to the user to indicate that he connected the wrong type of data to a side of a node, e.g., by breaking the wire that the user connected to the node. There is no teaching at all regarding programmatically generating graphical source code for the node, and there certainly is no teaching regarding programmatically generating graphical source code that implements specified functionality for the node.

Taking claim 10 as another exemplary claim, the claim recites the method of claim 1, further comprising in part, "receiving user input requesting to provide configuration information for the node". The Office Action relies on block 304 of FIG. 8 to teach this element of claim 10. However, as described in Col. 8, lines 18 – 20, block 304 simply refers to the user providing input to select the appropriate node icons to display in the block diagram. There is no teaching at all regarding the user requesting to provide configuration information for a node. Claim 10 further recites, "displaying a graphical user interface (GUI) input panel in response to the user input requesting to provide configuration information for the node". Thomsen simply does not teach the concept of displaying a GUI input panel in response to user input requesting to provide configuration information for a node. The cited portion of Col. 4, lines 47 – 56 does not in any way teach this concept but instead relates generally to the user using a block diagram editor to create a graphical program.

Claim Rejections – 35 U.S.C. 103(a)

Claims 11 – 19 and 28 – 33 were rejected under 35 U.S.C. 103(a) as being unpatentable over Thomsen and U.S. Patent No. 5,301,301 to Kodosky et al. (hereinafter referred to as "Kodosky"). Applicant respectfully traverses this rejection.

Kodosky relates generally to a system and method for constructing a dataflow block diagram that utilizes built-in function icons that are polymorphic with respect to data type and with respect to data aggregation. Applicant submits that there are numerous elements of the present claims that are not taught by Thomsen or Kodosky, taken either singly or in combination.

As per claim 14, the claim recites in pertinent part, "receiving user input specifying an alias for at least one input terminal or output terminal of the node". The Office Action states that, "Thomsen teaches the configuring alias (names) for the node (col. 2, lines 34-37)." However, this is not what the claim recites. As noted above, what the claim actually recites is "receiving user input specifying an alias for at least one input terminal or output terminal of the node", not an alias for the node itself. A name or alias for a node is not at all the same thing as an alias for an input terminal or output terminal of a node. Furthermore, the node name taught in Thomsen does not even appear to be configurable as asserted by the Examiner. Thomsen simply states that, "the front side of each node is reserved for displaying the name of the node or the fraction performed by the node." There is no teaching or suggestion that the user can provide input to specify the name.

The claim further recites, "for each input terminal or output terminal for which an alias was specified, displaying the alias in the graphical program". The Office Action relies on Kodosky to teach this element of the claim. However, Kodosky does not teach displaying an alias for an input terminal or output terminal, where the alias was specified by user input. Moreover, the cited portion of Kodosky clearly teaches that front panel control or indicator names are displayed in another window, not in the graphical program. As stated in Col. 2, lines 34 – 37, "LabVIEW has a 'help' feature which displays in another window the icon connection diagram: the icon with short wires attached to each input and output and labeled with the name of the associated front panel control or indicator." Thus, the element of "displaying the alias in the graphical program" as recited in claim 14 is not taught by Kodosky.

Applicant thus submits that claim 14 is allowable. Inasmuch as claims 30 and 35 recite similar features as those of claim 14, Applicant submits that these claims are also allowable. Since claims 15 – 19 depend on claim 14 and claims 31 – 33 depend on claim 30, Applicant submits that these claims are also allowable for at least this reason. Additionally, Applicant submits that dependent claims 15 – 18 and 31 – 33 recite numerous additional limitations not taught or suggested by Thomsen or Kodosky, taken either singly or in combination.

CONCLUSION

Applicant submits the application is in condition for allowance, and an early notice to that effect is requested.

If any extensions of time (under 37 C.F.R. § 1.136) are necessary to prevent the above referenced application(s) from becoming abandoned, Applicant(s) hereby petition for such extensions. If any fees are due, the Commissioner is authorized to charge said fees to Meyertons, Hood, Kivlin, Kowert & Goetzel PC Deposit Account No. 50-1505/5150-48500/JCH.

Also enclosed herewith are the following items:

☒ Return Receipt Postcard
□ Request for Approval of Drawing Changes
□ Notice of Change of Address
□ Check in the amount of $ for fees ( ).
□ Other:

Respectfully submitted,

Jeffrey C. Hood
Reg. No. 35,198
ATTORNEY FOR APPLICANT(S)

Meyertons, Hood, Kivlin, Kowert & Goetzel PC
P.O. Box 398
Austin, TX 78767-0398
Phone: (512) 853-8800
Date: 7/19/2004